

CLAIMS

What is claimed is:

1. A system comprising:

a system memory;

a computer processing module, including:

a host processing element configured to perform a task;

a data-generating processing element configured to perform a subtask within the task, including:

logic configured to receive input data; and

logic configured to process the input data to produce output data, wherein an amount of output data is greater than an amount of input data, a ratio of the amount of input data to the amount of output data defining a decompression ratio,

wherein the output data generated by the data-generating processing element is not contained in the system memory prior to it being generated by the data-generating processing element;

a cache memory coupled to the data-generating processing element for receiving the output data;

a computer processing module interface for outputting the output data from the cache memory;

a communication bus;

a data processing module, including:

1 a data processing module interface coupling to the computer
2 processing module interface via the communication bus for receiving the
3 output data; and

4 a data processing engine for receiving and processing the output
5 data from the cache memory, wherein the data processing engine uses a
6 tail pointer to indicate a location within the cache memory from which it
7 has just retrieved data;

8 wherein, in a write streaming mode of operation, the computer processing module
9 is configured to allocate a portion of the cache memory for the purpose of receiving
10 streaming write output data from the data-generating processing element,

11 wherein, in the write streaming mode of operation, the system is configured to
12 forward output data from said allocated portion of the cache memory to the data
13 processing module rather than from the system memory, and

14 wherein the data processing module is configured to forward the tail pointer to a
15 cacheable address of the data-generating processing element, the tail pointer informing
16 the data-generating processing element of the location within the cache memory from
17 which the data processing module has just retrieved data.

18
19 2. A system according to claim 1, wherein the host processing element comprises
20 a thread implemented on a computer processing unit, and the data-generating processing
21 element comprises a thread implemented on the same computer processing unit or
22 implemented on another computer processing unit.

23
24 3. A system according to claim 1, further comprising plural host processing
25 elements.

1
2 4. A system according to claim 3, wherein the plurality host processing elements
3 comprise a plurality of respective threads implemented on at least one computer
4 processing unit.

5
6 5. A system according to claim 1, further comprising plural data-generating
7 processing elements.

8
9 6. A system according to claim 5, wherein the plurality of data-generating
10 processing elements comprises plural respective threads implemented on at least one
11 computer processing unit.

12
13 7. A system according to claim 1, wherein the host processing element and the
14 data-generating processing element each perform functions that are statically allocated.

15
16 8. A system according to claim 1, wherein the host processing element and the
17 data-generating processing element each perform functions that are dynamically
18 allocated.

19
20 9. A system according to claim 1, further comprising plural data-generating
21 processing elements, wherein each of the plural data-generating processing elements is
22 coupled to the cache memory.

1 10. A system according to claim 1, wherein the data-generating processing
2 element includes an L1 cache, and said cache memory of the computer processing
3 module is an L2 cache.
4

5 11. A system according to claim 10, wherein, in a read streaming mode of
6 operation, the computer processing module is configured to provide the input data by
7 forwarding the input data to the L1 cache of the data-generating processing element, by
8 bypassing the L2 cache.
9

10 12. A system according to claim 10, wherein, in the write streaming mode of
11 operation, the computer processing module is configured to forward the output data to the
12 L2 cache by bypassing the L1 cache.
13

14 13. A system according to claim 1, wherein the cache memory is an n-way set-
15 associative cache, and wherein the portion is allocated by locking at least one set of the n-
16 way set-associative cache.
17

18 14. A system according to claim 1, wherein the allocated portion of the cache
19 memory forms at least one FIFO buffer that couples the data-generating processing
20 element to the data processing module.
21

22 15. A system according to claim 14, wherein the system is configured to wrap
23 within said at least one FIFO buffer by using a middle section of an address to index said
24 at least one FIFO buffer, wherein an upper section and a lower section of the address are
25 ignored by the system.

1
2 16. A system according to claim 1, wherein the data processing module is
3 configured to process output data received from the cache memory using a modified
4 direct memory access (DMA) protocol.
5

6 17. A system according to claim 1, wherein the computer processing module is
7 configured to maintain a cache line state of dirty after accessing a cache line.
8

9 18. A system according to claim 1, wherein the decompression ratio is at least 1
10 to 10.
11

12 19. A system according to claim 1, wherein the decompression ratio is at least 1 to
13 100.
14

15 20. A system according to claim 1, wherein the decompression ratio is at least 1 to
16 1000.
17

18 21. A system according to claim 1, wherein the data-generating processing
19 element is configured to dynamically vary the ratio of decompression during its operation
20 in response to at least one criterion.
21

22 22. A system according to claim 21, wherein said at least one criterion is depth of
23 scene associated with an object in a scene.
24
25

1 23. A system according to claim 1, wherein the logic for processing the input data
2 further comprises logic configured to execute a dot product operation upon receipt of a
3 dot product instruction using an array of structures computational technique.

4
5 24. A system according to claim 1, wherein the logic for processing the input data
6 further comprises logic for compressing data from a first information content amount to a
7 second information content amount to provide the output data, wherein the first
8 information content amount is greater than the second information content amount.

9
10 25. A system according to claim 1, wherein the task performed by the host
11 processing element pertains to a graphics processing task, and wherein the subtask
12 performed by the data-generating processing element pertains to the generation of
13 geometry data.

14
15 26. A system according to claim 25, wherein the task performed by the host
16 processing element pertains to high level aspects of a three dimensional game
17 application.

18
19 27. A system according to claim 25, wherein the logic for processing input data
20 comprises procedural geometry logic configured to transform the input data into the
21 output data, wherein the output data comprises a set of vertices.

22
23 28. A system according to claim 25, wherein the logic for processing input data
24 comprises a higher order surface tessellation engine configured to transform information
25 expressed in a higher order surface into output data comprising a set of vertices.

1
2 29. A system comprising:
3 a system memory;
4 a host processing element configured to perform a task;
5 a data-generating processing element configured to perform a subtask within the
6 task, including:
7 logic configured to receive input data; and
8 logic configured to process the input data to generate output data,
9 wherein an amount of output data is greater than an amount of input data,
10 a ratio of the amount of input data to the amount of output data defining a
11 decompression ratio,
12 wherein the output data generated by the data-generating
13 processing element is not contained in the system memory prior to it being
14 generated by the data-generating processing element;
15 a cache memory for storing the output data generated by the data-generating
16 processing element in an allocated portion thereof;
17 a communication bus;
18 a data processing engine configured to retrieve the output data from the cache
19 memory via the communication bus, and to process the output data, wherein the data
20 processing engine uses a tail pointer to indicate a location within the cache memory from
21 which it has just retrieved data; and
22 a tail pointer updating mechanism configured to provide tail pointer updates to a
23 cacheable address of the data-generating processing element via the communication bus.
24
25

1 30. A method for processing data in a system including a host processing element,
2 a data-generating element, and a data processing engine, wherein the host processing
3 element and the data-generating element are coupled to the data processing engine via a
4 communication bus, comprising:

5 performing a task in a host processing element, the task requiring the execution of
6 a subtask as a part thereof;

7 performing the subtask in a data-generating processing element when commanded
8 by the host processing element, the performing of the subtask including:

9 receiving input data; and

10 processing the input data to produce output data, wherein an
11 amount of output data is greater than an amount of input data, a ratio of
12 the amount of input data to the amount of output data defining a
13 decompression ratio,

14 wherein the output data generated by the data-generating
15 processing element is not contained in a system memory prior to it being
16 generated by the data-generating processing element;

17 buffering the output data in an allocated portion of a cache memory;

18 retrieving, by a data processing engine, the output data from the cache memory
19 via the communication bus, rather than the system memory; and

20 processing the retrieved output data in the data processing engine, wherein the
21 data processing engine uses a tail pointer to indicate a location within the cache memory
22 from which it has just retrieved data; and

23 forwarding the tail pointer to a cacheable address of the data-generating
24 processing element, the tail pointer informing the data-generating processing element of
25

1 the location in the cache memory from which the data processing engine has just
2 retrieved data.

3
4 31. A method according to claim 30, wherein the host processing element
5 comprises a thread implemented on a computer processing unit, and the data-generating
6 processing element comprises a thread implemented on the same computer processing
7 unit or implemented on another computer processing unit.

8
9 32. A method according to claim 30, further comprising plural host processing
10 elements.

11
12 33. A method according to claim 32, wherein the plurality host processing
13 elements comprise a plurality of respective threads implemented on at least one computer
14 processing unit.

15
16 34. A method according to claim 30, further comprising plural data-generating
17 processing elements.

18
19 35. A method according to claim 34, wherein the plurality of data-generating
20 processing elements comprises plural respective threads implemented on at least one
21 computer processing unit.

22
23 36. A method according to claim 30, wherein the host processing element and the
24 data-generating processing element each perform functions that are statically allocated.
25

1 37. A method according to claim 30, wherein the host processing element and the
2 data-generating processing element each perform functions that are dynamically
3 allocated.

4
5 38. A method according to claim 30, further comprising plural data-generating
6 processing elements, wherein each of the plural data-generating processing elements is
7 coupled to the cache memory.

8
9 39. A method according to claim 30, wherein the data-generating processing
10 element includes an L1 cache, and said above-referenced cache memory is an L2 cache.

11
12 40. A method according to claim 39, wherein, in a read streaming mode of
13 operation, the data-generating processing element receives the input data by forwarding
14 the input data to the L1 cache of the data-generating processing element, by bypassing
15 the L2 cache.

16
17 41. A method according to claim 39, wherein, in a write streaming mode of
18 operation, the data-generating unit provides the output data by forwarding the output data
19 to the L2 cache by bypassing the L1 cache.

20
21 42. A method according to claim 30, wherein the cache memory is an n-way set-
22 associative cache, and wherein the portion is allocated by locking at least one set of the n-
23 way set-associative cache.

1 43. A method according to claim 30, wherein the allocated portion of the cache
2 memory forms at least one FIFO buffer that couples the data-generating processing
3 element to the data processing engine.

4
5 44. A method according to claim 43, further comprising wrapping within said at
6 least one FIFO buffer by using a middle section of an address to index said at least one
7 FIFO buffer, wherein an upper section and a lower section of the address are ignored by
8 the method.

9
10 45. A method according to claim 30, wherein the data processing engine
11 processes output data received from the cache memory using a modified direct memory
12 access (DMA) protocol.

13
14 46. A method according to claim 30, further comprising maintaining a cache line
15 in a state of dirty after accessing a cache line.

16
17 47. A method according to claim 30, wherein the decompression ratio is at least 1
18 to 10.

19
20 48. A method according to claim 30, wherein the decompression ratio is at least 1
21 to 100.

22
23 49. A method according to claim 30, wherein the decompression ratio is at least 1
24 to 1000.

25

1 50. A method according to claim 30, wherein the performing of the subtask
2 includes dynamically varying the ratio of decompression during operation of the data-
3 generating processing element in response to at least one criterion.

4
5 51. A method according to claim 50, wherein said at least one criterion is depth of
6 scene associated with an object in a scene.

7
8 52. A method according to claim 30, wherein the performing of the subtask
9 involves executing a dot product operation upon receipt of a dot product instruction using
10 an array of structures computational technique.

11
12 53. A method according to claim 30, wherein the performing of the subtask
13 involves compressing data from a first information content amount to a second
14 information content amount to provide the output data, wherein the first information
15 content amount is greater than the second information content amount.

16
17 54. A method according to claim 30, wherein the task performed by the host
18 processing element pertains to a graphics processing task, and wherein the subtask
19 performed by the data-generating processing element pertains to the generation of
20 geometry data.

21
22 55. A method according to claim 54, wherein the task performed by the host
23 processing element pertains to high level aspects of a three dimensional game
24 application.
25

1 56. A method according to claim 54, wherein the processing of input data
2 comprises performing procedural geometry to transform the input data into the output
3 data, wherein the output data comprises a set of vertices.

4
5 57. A method according to claim 54, wherein the processing of input data
6 comprises performing higher order surface tessellation to transform information
7 expressed in a higher order surface into output data comprising a set of vertices.